

me



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/410,334	10/01/1999	AUBREY MCAULEY	ADHE1100	6725

30076 7590 01/30/2004

BROWN RAYSMAN MILLSTEIN FELDER & STEINER, LLP
SUITE 711
1880 CENTURY PARK EAST
LOS ANGELES, CA 90067

EXAMINER

CHAVIS, JOHN Q

ART UNIT	PAPER NUMBER
----------	--------------

2124

DATE MAILED: 01/30/2004

14

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/410,334

Applicant(s)

MCAULEY, AUBREY

Examiner

John Chavis

Art Unit

2124

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 06 October 2003.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-53 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-53 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. §§ 119 and 120

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
* See the attached detailed Office action for a list of the certified copies not received.
- 13) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.
a) ☐ The translation of the foreign language provisional application has been received.
- 14) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892) 4) ☐ Interview Summary (PTO-413) Paper No(s). _____
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948) 5) ☐ Notice of Informal Patent Application (PTO-152)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ 6) ☐ Other: _____

DETAILED ACTION

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

2. Claims 1-8, 11-18, 20-21, 25-40, 42-44, 47, 50-51 and 53 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lewandowski (PTO-1449 reference U) in view of the applicant's background acknowledgement of prior art and further in view of Ferrel et al. (6,199,082). The applicant's invention pertains to a method and system of generating computer applications from a framework. The invention, specifically (for example, claim 1) merely appears to consist of having a program developer enter selective data (form, content and functionality) in separate objects. The features of the applicant's claims are now presented in aside-by-side manner with the teachings of Lewandowski/Admitted prior art in view of Ferrel.

<u>Claims</u>	<u>Lewandowski/Admitted art/Ferrel</u>
1. A method for generating computer application on a host system in an arbitrary object framework that separates a content of said computer application, a form of said computer application and a functionality of said computer applications, said method comprising:	See figure 8 (host) and fig. 5 (IDL form or interface), functionality (page 14, which is separate from the interface (1st complete paragraph)), and content (which is considered merely data), Tier three on page 22, all in the Lewandowski reference. The applicant argues that Lewandowski/Admitted art does not Teach the feature of separating form/function; which is not considered the case; however, assuming the feature is not present, the feature is taught by Ferrell, see the abstract. Ferrell teach the feature to reduce the size of downloads when the format (form) remains the same. He separates form from content (functionality and content). Note all art discussions above and below are in view of Lewandowski unless listed otherwise.

Art Unit: 2124

creating arbitrary objects with corresponding arbitrary names of various object types for generating said content of said computer application, said form of said computer application, and said Said functionality of said and said functionality of said application;

Objects in an object oriented environment are inherently arbitrary to provide for the specific details of the application with corresponding arbitrary names (see the naming service on page 13, Lewandowski) specified to identify and enable access to the specific objects, see again the abstract. Furthermore, see Lewandowski's figure 10 on page 22, with it's BLO to generate content (data), Presentation Objects to represent form and its BPO to represent functionality. Although, it is considered inherent that content is a separate object in Lewandowski's system, the feature may be argued since it is not specifically stated. However, the applicant admits that the feature existed in the prior art on page 3 lines 18-29 to help minimize dependencies between interface designs and the functions they access. Furthermore, as indicated above, Ferrel provides for the feature, see also Ferrel's col. 2 line 35-col. 3 line 20 and col. 5 lines 21-59. Ferrel teach the use of a style sheet object (form-col. 32 lines 5-14), content (content-col. 32 lines 38-62), and search objects (functionality-col. 39 lines 53). Note also that each of Ferrel items are separate to enable control over the design process, see col. 44 lines 14-16. Ferrel's system also enable process (functionality or functions) to make requests to data (content) centers, which further indicates that the two are separate, col. 62 lines 12-38.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of the invention to modify Lewandowski's system with the applicant's admitted prior art feature of separating content from form and functionality for the same reasons as they were utilized in the prior art and also Lewandowski/Admitted art with the feature of separating the form from the content (content/data and functionality/methods/processes). It would also, as indicated above merely be a design choice to separate the form and function objects. However, the modifications in view of Ferrel would have been obvious

Art Unit: 2124

to a person of ordinary skill in the art for the same reasons they are taught by Ferrel, to simplify downloads when possible.

managing said arbitrary objects in an object library; and

furthermore, objects and functions (functionality) are provided names to enable communication (managing) between objects, via CORBA, page 23. The library feature (framework) enables quick and easy reusability, see the Conclusions on page 26 and again see the Introduction for the library feature.

Deploying said arbitrary objects from Said object library into a design Framework to create said computer.

see section 5.1.3 on page 16.

2. The method of claim 1, wherein said computer application is a web Site.

Lewandowski provides his system as a client/server system, which inherently represents any system that makes requests and provides responses, such a web site.

3. The method of claim 1, wherein said various object types comprise text file pointers.

this feature is considered inherent to enable access to specific data items in a database such as "shippable products, see section 5.2.

4. The method of claim 1, wherein said various object types comprise binary file pointers.

see again the rejection of claim 3.

5. The method of claim 1, wherein said various object types comprise executables.

see again the BPO's in claim 1.

6. The method of claim 1, wherein said various object types comprise shell commands.

this feature is considered inherent via the search commands and the Access commands, via the Presentation Objects indicated in claim 1.

7. The method of claim 1, wherein specifically said various object types comprise remote procedure calls.

see again the Conclusion, the first paragraph.

8. The method of claim 1, wherein said various object types comprise global variables.

this feature is considered inherent in object oriented Systems to identify variables that are generic (common features) to all subclasses and therefore not

11. The method of claim 1, wherein said various object types comprise local variables.
subclass.
12. The method of claim 1, wherein said various object types comprise local objects and global parent objects.
13. The method of claim 12, wherein said local objects can override said global parent objects.
14. The method of claim 12, wherein said local objects inherit data from said global parent objects.
15. The method of claim 1, wherein said local objects inherit capabilities from said global parent objects.
16. The method of claim 1, further comprising arbitrary objects globally.
17. The method of claim 1, further comprising arbitrary objects locally.
18. The method of claim 1, wherein the step of managing said arbitrary Objects further comprises revision tracking.
20. The method of claim 1, wherein the step of managing further comprises using signoff.
access
21. The method of claim 1, wherein said arbitrary objects can be accessed and deployed into said design framework using said corresponding arbitrary names.
25. The method of claim 1, further Comprising generating arbitrary
- changed In Subclasses. The data that is not overridden is inherently global to the subclass that does not provide it's own definition.
- see the discussion in claim 8, the data that is overridden or new in subclasses are local to the
- see the rejections of claims 8 and 11.
- see again the rejections of claims 8 and 11.
- see the rejections of claims 8 and 11.
- see again the rejection of claims 8 and 11.
- see again the rejection of claims 8 and 11.
- see again the rejection of claims 8 and 11.
- see the last sentence of the first paragraph of section 3. 1.
- This feature is considered inherent via the sign in feature via password Verifications to ensure proper to stored information in a client/server environment, see section 2.
- see the rejection of claim 1.
- This feature is considered inherent in CORBA systems.

Art Unit: 2124

objects in a programming language that is compatible or Supported by said host system.

The features of claims 26-32 are taught via the rejections of claims 2-8.

As per claims 33-36, see the rejections of claims 11-14.

In reference to claims 37-40, see claims 15-18.

The features of claims 42-44 are taught via claims 20-22.

As per claim 47, see the rejection of claim 25.

In reference to claim 50-51, see the dynamic construction on page 12.

The features of claim 53 are indicated via the object containment feature on page 15.

Claim Rejections - 35 USC § 103

3. Claims 9-10, 19 and 22-24, 41, 45-46, 48-49, 52 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lewandowski in view of the applicant's admitted prior art and Ferrel, as cited above, and further in view of Gish (6,253.2S2) who teach providing caching to speed access time, using rollback to compensate for errors in subclasses via his event handling means and swapping objects when a different type is required or needed.

claims

Lewandowski/Admitted art/Ferrel/Gish

9. The method of claim 1, wherein said various object types comprise cached executables.

Lewandowski does not specifically teach the use of cached executables; However, he teach the use of executables as indicated in the rejection of claim 5. Gish teach the use of a cache, see the abstract, to speed up data access time and it would have been obvious to a person of ordinary skill in the art at the time of the invention to modify Lewandowski's system with Gish's caching of data to increase execution speed.

10. The method of claim 1, wherein said various object types comprise cached database queries.

in reference to the database queries, this is the essence of client/Server systems, see section 2.2. In reference to the caching of data see claim 9.

Art Unit: 2124

19. The method of claim 1, wherein the step of managing said arbitrary objects further comprises using rollback.

Although Lewandowski does not specifically indicate that rollback is used, the feature is considered a standard feature for correcting errors to ensure that previous updates are not lost. Gish teaches handling events that occur as a program executes. Therefore, it would have been obvious to a person of ordinary skill in the art at the time of the invention to utilize the feature of handling events taught by Gish (col. 36 line 50-col. 38 line 52) in Lewandowski's system when the validator, col. 27 lines 14-15, encounters errors to ensure that previously validated data is not lost. Gish enables the creation of specific handlers and therefore it would have been obvious to a person of ordinary skill in the art to utilize a rollback to reduce update time.

22. The method of claim 1, further comprising swapping an arbitrary object of one type with an arbitrary object of another type.

Lewandowski does not specifically teach that objects of one type are swapped for objects of another type; however, he does indicate that objects are self-managed as needed from the problem domain, section 3.2, and that they function in multiple environments, section 3.3. Therefore, it would have been obvious to a person of ordinary skill in the art at the time of the invention to select different types when different types are needed to solve a specific problem.

23. The method of claim 1, further comprising caching objects.

see the rejections of claim 9 and 10.

24. The method of claim 23, wherein the step of caching objects further comprises specifying some elements of an arbitrary object to be dynamic elements and specifying some elements of said arbitrary to be static elements.

see the rejections of claims 9 and 10 in view of see the second complete paragraph of page 12.

In reference to claim 4, see the rejection of claim 19.

As per claims 45-46, see the rejection of claims 23-24.

Art Unit: 2124

In reference to claim 48-49, see claims 9-10.

The features of claim 52 are taught via claim 24.

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to John Chavis whose telephone number is (703) 305-9665. The examiner can normally be reached on 8:30 - 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (703) 305-9662. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.



JC
March 26, 2003



JOHN CHAVIS
PATENT EXAMINER
ART UNIT 2124